

REMARKS

Claims 11-22, 24, and 25 are pending.

Claims 1-10, 23, and 26-36 are withdrawn.

Claims 11, 12, 13, and 21 have been amended.

Claims 37 and 38 have been added.

Claim Objection

Claim 12 is objected to for use of the acronym “ID”. “ID” has been replaced with “identification”. Applicants respectfully request withdrawal of the objection.

Claim Rejections – 35 U.S.C. § 101

Claims 11 and 21 stand rejected under 35 U.S.C. § 101 as directed to non-statutory subject matter.

Claim 11 has been amended to specifically recite that the “identifying”, both “including”, and “representing” elements are performed by a computer system.

Claim 21 has been amended to specifically recite that the instructions are executable by a processor.

Accordingly, Applicants respectfully request withdrawal of the rejections.

Claim Rejections – 35 U.S.C. § 112, first para.

Claims 11-22, 24, and 25 stand rejected under 35 U.S.C. § 112, first paragraph, as failing to comply with the enablement requirement.

A.

The Office Action specifically cites a lack of enabling disclosure for the “establishing” and “including the leaf node” elements. Applicants respectfully submit that the specification provides enabling disclosure in, for example, p. 5, line 16 - p. 6, line 8, page 8, line 23 - p. 9, line 2, page 9, line 20 - page 10, line 20, page 19, line 20, p. 20, line 13, page 23, line 24 - page 26, line 24, and Figs. 4c and 4d.

In general terms, a custom browse hierarchy is generated using a primary hierarchy. The primary hierarchy has leaf nodes and each leaf node may have one or more ancestor nodes. Each ancestor node has a constraint, such as Product Type = Software and Vendor = Microsoft. By logically aggregating the constraints of each ancestor node of a leaf node, a method can determine whether or not a particular item meets the constraints. For example, an item Microsoft Word meets the constraints, but an item Linux operating system does not. If a rule is identified whose constraints are met by an item, the leaf node and ancestor nodes associated with the identified rule are included in the custom browse hierarchy. Thus, a custom browse hierarchy, which is a pared version of the primary hierarchy, is generated. Because the custom browse hierarchy is a “pared” version of the primary hierarchy, at least one of the nodes in the primary hierarchy is not included in the custom browse hierarchy.

Claim 11 recites:

establishing a set of rules for at least the ancestor nodes of the primary hierarchy, wherein each rule in the set of rules is associated with one of the leaf nodes and each ancestor node of the leaf node and each rule comprises a logical aggregation of constraints specified by at least each ancestor node of the leaf node;

According to the Specification, “for each leaf node of the primary hierarchy, a search rule is derived that comprises an aggregation of constraints specified by the leaf node and its ancestors.” Present Application, p. 5, lines 18-20. The Specification also recites for an exemplary embodiment:

As one navigates down a path of connected nodes in one embodiment of such a hierarchy, the scope of items that are defined by (i.e. fall under) a node in the path tends to become progressively more narrow. This is because the constraints imposed on attribute values at any node in the hierarchy include all of the constraints specified by the node as well as its ancestor nodes. These constraints are implicitly ANDed together logically as an aggregation of such constraints as one navigates or browses through a path in the hierarchy. Thus, the number of items meeting the aggregation of constraints typically decreases as one travels down the browse path. The items that fall under a leaf node in the hierarchy can be thought of as only those items in the database that meet the aggregation of constraints on attributes and their values inherited from each of the nodes in the path all the way up to the root. *Id.*, p. 8, line 23-p. 9, line 2.

The Specification provides exemplary rules on p. 19, line 20-p. 20, line 13.

Claim 11 also recites:

for each leaf node in the primary hierarchy, including the leaf node in the custom browse hierarchy if the rule associated with the leaf node is included in the subset of the set of rules;

The Specification states:

Likewise, for each custom version of the catalog, a custom version of the primary browse hierarchy may also generated based on the primary browse hierarchy. In one embodiment, the process of generating customized browsing hierarchies starts out by initiating a query of the catalog database for each of the leaf nodes of the primary hierarchy. The query for each leaf node is formulated by first by aggregating all of the constraints specified by each node of the hierarchy in the path between the leaf node and the root (i.e. that is an ancestor of the leaf node). An “include” rule is then established dictating that all items should be included that meet the aggregation of constraints. The database query is then derived from this rule and uses the results from the rule set searches stored in the subset ID table to return a set of subset identifiers representing all of the custom subsets that include at least one item SKU that meets the aggregated constraints of the rule.

In one embodiment, the query performs two functions at once. First, it searches the database until it locates an item SKU that meets the constraints, and it then joins this result with the subset ID table to obtain the subset (i.e. rule set) identifiers for all subsets that contain that SKU. This process continues until all SKUs in the database that meet the constraints have been processed in this manner. The result of the complete query is a set of all subset identifiers for catalog subsets that contain at least one SKU that meets the constraints, and this set is stored for each leaf node for use in generating the custom browse hierarchies.

Based on the search results temporarily stored for each leaf node of the primary hierarchy, the method for generating custom browse hierarchies then generates a custom hierarchy for each custom subset (i.e. catalog). Each custom hierarchy is generated for a particular subset by cloning (i.e. including in the custom hierarchy) only those leaf nodes (and their ancestors) for which there is at least one item SKU in the catalog subset. Put another way, the custom browse hierarchy generated for a particular custom catalog subset retains only those leaf nodes (and their ancestors) from the primary hierarchy for which the set of stored subset IDs resulting from the search for each leaf node described above includes the subset ID for the subset for which the browse hierarchy is being generated. This process is repeated for each custom catalog subset and a custom browse hierarchy is stored for each custom catalog and is identified with the same rule set identifier (i.e. subset ID) with which the custom subset is identified. *Id.*, p. 9, line 20-p. 10, line 20.

The Specification provides a more specific example described in page 23, line 24-page 26, line 24 and Figs. 4c and 4d.

B.

The Examiner has rejected claim 11 for alleged redundancy in the “including” elements of claim 11. Applicants respectfully submit that the elements were not redundant. However, Applicants have amended the two “including” elements for clarity. The claims have not been amended to recite “including each leaf node” and “including each ancestor node” because inclusion is conditional. However, the conditional aspect is not indefinite or ambiguous because open-ended conditions are clearly recited, i.e. “ including the leaf node in the custom browse hierarchy if the rule associated with the leaf node is included in the subset of the set of rules” and “including the ancestor node in the custom browse hierarchy if at least one leaf node of the ancestor node is included in the custom browse hierarchy.”

C.

The Examiner states that “computer-readable storage medium and program instructions stored on the storage medium are not found in Applicants’ specification.” Applicants respectfully disagree. For example, originally filed claim 21 recites “a computer readable storage medium; and program instructions stored on said storage medium.” See Manual of Patent Examining Procedure § 608.01(I), “In establishing a disclosure, applicant may rely not only on the description and drawing as filed but also on the original claims if their content justifies it.”

D.

The Examiner also states that ““pared version of the primary hierarchy” is not found in Applicants’ specification.” Applicants respectfully submit that “the custom browse hierarchy represents a pared version of the primary hierarchy” as recited in claims 11 and 21 is clearly supported by the specification. Applicants respectfully disagree. Applicants respectfully submit that support can be found on, for example, p. 1, lines 5-7 (“the customized browse hierarchies being pared down in scope from a primary hierarchy in accordance with rules-based searches of a central catalog database maintained by the seller”), p. 5, lines 6-7 (“Each custom browse hierarchy would have a scope that is pared down from the primary hierarchy”), p. 12, lines 22-24

(“Moreover, the browse hierarchy is pared down in scope from the most general primary hierarchy to a scope of items that is coextensive with (or at least does not exceed) the scope of items the particular buyer is permitted to see.”), and p. 16, lines 26-28 (“the custom browse hierarchy is pared down to have a scope that is coextensive with that subset of items is completely transparent to the buyers in both the extranet and non-extranet contexts.”). The Examiner’s particular definition of “pare” on page 4 of the Office Action could imply a particular method of paring. Generating the custom browse hierarchies is defined by the claims. Applicants respectfully disavow the Examiner’s definition to the extent the Examiner’s definition conflicts with the invention as claimed.

Accordingly, Applicants respectfully request withdrawal of the rejections.

Applicants respectfully submit that the invention is defined by the claims and not limited by specific examples set forth in the Specification or specific examples set forth in this Response.

Claim Rejections – 35 U.S.C. § 112, second para.

Claims 11 and 21 stand rejected under 35 U.S.C. § 112, second paragraph, as being indefinite.

A.

The Examiner has rejected the claims for use of the term “if any” throughout claims 11 and 12. Claims 11 and 21 have been amended, in part, to remove the phrase “if any”. The removal of the phrase “if any” does not mean that each leaf node in the primary hierarchy must have an ancestor node. However, the specific leaf nodes in the open-ended claims 11 and 21 have ancestor nodes.

B.

The Examiner has rejected claims 11 and 21 because of the use of ‘including and otherwise excluding’. Applicants respectfully disagree that the claims were either vague or unclear. Because of the open-ended nature of claims 11 and 21 inclusion of certain leaf nodes does not automatically require exclusion of all other leaf nodes. Thus, the ‘including and otherwise excluding’ was not unclear or vague. Applicants have amended claims 11 and 21 to

remove the 'excluding' limitation from claims 11 and 21. However, Applicants have added a corresponding limitation in new claims 37 and 38. Because the custom browse hierarchy is a "pared" version of the primary hierarchy, at least one of the nodes in the primary hierarchy is not included in the custom browse hierarchy.

C.

The Examiner has rejected claims 11 and 21 for omitting "custom browse". The specific location of the omission was not provided. Applicants have amended claim 11 to recite "generating each of the plurality of custom browse hierarchies". Applicants could not locate a corresponding location in claim 21.

Accordingly, Applicants respectfully request withdrawal of the rejections.

Claim Rejections – 35 U.S.C. § 103

I.

Claims 11 and 21 stand rejected under 35 U.S.C. § 103 as unpatentable over U.S. Patent No. 5,933,599 to Nolan (hereinafter "*Nolan*") in view of U.S. Patent No. 6,460,025 to Fohn et al. (hereinafter "*Fohn*"). Applicants respectfully traverse the rejection.

Referring to Figure 2 of *Nolan*, *Nolan* teaches that "the on-line network is organized into a hierarchical data structure 200 similar to a hierarchical tree." *Nolan*, col. 6, lines 47-49. "Each item in the structure is referred to as a "node." *Id.*, line 49. The Examiner cites col. 8, lines 1-45 and Fig. 2 (204) and (210) of *Nolan*. Applicants respectfully submit that col. 8, lines 1-45 of *Nolan* neither teach nor suggest "establishing a set of rules for at least the ancestor nodes of the primary hierarchy, wherein each rule in the set of rules is associated with one of the leaf nodes and each ancestor node of the leaf node and each rule comprises a logical aggregation of constraints specified by at least each ancestor node-of the leaf node" as required by claims 11 and 21.

Nolan teaches that each node 204, 206, and 208 has a set of node properties. See, e.g. *Nolan*, col. 8, lines 1-5. However, having properties neither teaches nor suggests "establishing a set of rules for at least the ancestor nodes of the primary hierarchy, wherein each rule in the set of rules is associated with one of the leaf nodes and each ancestor node of the leaf node and each

rule comprises a logical aggregation of constraints specified by at least each ancestor node-of the leaf.” Claims 11 and 21. Applicants respectfully submit that having nodes in a hierarchy and specifying properties for the nodes neither teaches nor suggests establishing a set of rules where “each rule comprises a logical aggregation of constraints specified by at least each ancestor node-of the leaf.” *Id.*

Fohn relates to a method, system, and computer readable code for improving user exploration (i.e. navigation and browsing) through hierarchies of information. *Fohn*, Abstract. The Examiner cites *Fohn* as teaching identifying a rule subset of the set of rules, wherein each rule in the rule subset has constraints that are met by at least one of the items in the unique subset, etc. Office Action, p. 7. Applicants respectfully submit that any rules taught by *Fohn* are not the specific types of rules required by claims 11 and 21 as previously identified, e.g. “each rule comprises a logical aggregation of constraints specified by at least each ancestor node-of the leaf.”

Additionally, Applicants respectfully submit that the combination of *Nolan* and *Fohn* would **not** teach or suggest “custom browse hierarchies” of a primary hierarchy wherein “the custom browse hierarchy represents a pared version of the primary hierarchy.” *Fohn* specifically states that, “The present invention addresses the inadequacies of hierarchical navigation and browsing by combining entity relevance with node-based forward-checking to implement an intelligent hierarchical exploration scheme.” *Fohn*, col. 8, lines 19-22. *Fohn* continues by describing how the “invention facilitates and implements intelligent hierarchical exploration.” *Id.*, col. 9, lines 1-2. Thus, a combination of *Nolan* and *Fohn* would actually teach against generating custom browse hierarchies because the combination of *Nolan* and *Fohn* would *arguendo* provide a better way of exploring a primary hierarchy **without** creating a custom browse hierarchy which “**represents a pared version of the primary hierarchy**”.

Applicants respectfully submit that there is no teaching or suggestion in the combination of *Nolan* and *Fohn* of, for example, the specific rules of claims 11 and 21 and no teaching or suggestion that *Fohn* would be used to modify *Nolan* to create custom browse hierarchies which “**represents a pared version of the primary hierarchy**” as required by claims 11 and 21. Applicants respectfully submit that the combination of *Nolan* and *Fohn* would keep the primary hierarchy intact because of the presumably intelligent hierarchical exploration scheme.

Accordingly, for at least the foregoing reasons Applicants respectfully request withdrawal of the rejection of claims 11 and 21.

II.

Claims 12-20, 22, 24, and 25 stand rejected under 35 U.S.C. § 103 as unpatentable over *Fohn* as applied to claims 11 and 21, and further in view of U.S. Patent No. 6,466,918 to Spiegel et al. (hereinafter “*Spiegel*”). Applicants respectfully traverse the rejection.

Spiegel relates to “A computer-implemented system and method are provided for identifying popular nodes within a browse tree or other hierarchical browse structure based on historical actions of online users.” *Spiegel*, Abstract.

Claims 12-20, 22, 24, and 25 depend directly or indirectly from claims 11 or 21. Accordingly, for at least the foregoing reasons set forth with regard to *Nolan* in view of *Fohn*, Applicants respectfully request withdrawal of the rejection of claims 12-20, 22, 24, and 25.

CONCLUSION

The application is believed to be in condition for allowance and a notice to that effect is solicited. Nonetheless, should any issues remain that might be subject to resolution through a telephonic interview, the examiner is requested to telephone the undersigned at 512-338-9100.

CERTIFICATE OF TRANSMISSION

I hereby certify that on August 18, 2009, this correspondence is being transmitted via the U.S. Patent & Trademark Office's electronic filing system.

/Kent B. Chambers/

Respectfully submitted,

/Kent B. Chambers/

Kent B. Chambers
Attorney for Applicant(s)
Reg. No. 38,839